## REMARKS/ARGUMENTS

Reconsideration of this application is respectfully requested.

The Examiner is thanked for a very helpful interview with the undersigned and inventor Lloyd on November 26, 2008. As the Examiner then agreed, if the present response does not place this entire application in allowable condition, the undersigned will be telephoned for any further discussion/amendment required to allow this application.

The following remarks include a summary of comments made during the interview – and attached hereto is a copy of papers used during the interview (a copy of which was also left with the Examiner at the conclusion of the interview):

- Attachment 1: a 9-page Power Point® presentation identifying many novel aspects of the now claimed invention, including advantages of several features of the exemplary embodiment *vis-à-vis* the cited Harvey prior art; and

- Attachment 2: a 12-page presentation of new claims 139-223 discussed during the interview.

The Examiner is thanked for a helpful "response to arguments" section in the outstanding office action bridging pages 2-4. Based on a study of the Examiner's comments, as well as consultations with inventor Lloyd, the undersigned has now come to a better appreciation of some of the more salient novel and advantageous distinctions between the applicants' invention and the Harvey prior art teaching. Accordingly, based on this further study, a completely new set of claims 139-223 is herewith submitted. It will be noted that these new claims include only two independent claims: method claim 139 and a respectively corresponding apparatus claim 182.

In the prior art Harvey approach, the directory system provided only a relatively "dumb" data base which merely returned fields of data in response to an application module request. In effect, the data base was merely a "bucket" of data items. Any coherent "intelligent" decision-making concerning content of such data items was left to the intelligent application module making the data request.

As a result, each application module was required to make numerous data requests in order to assemble data items required for making some intelligent decision (e.g., is the person signing-on to a given terminal actually authorized to receive certain requested services or not?). Furthermore, each different application module must also request the same or similar data items so as to make the same or similar intelligent decisions.

Such prior art arrangements, therefore, require duplicitous functionality at many different application modules and typically numerous data requests/responses by each for each transaction tying up excessive data communication bandwidth and taking longer than desirable from an overall system viewpoint. For example, in some systems, there may literally be tens of millions of customers and several thousands of those customers may almost simultaneously be logging in and requesting access to certain services. The effective load (both from a processing standpoint and from a data communication standpoint) for directory services can thus become enormous and require relatively lengthy delays in granting authorized customers access to services and the like.

By contrast, the applicants' novel approach is to, in effect, (a) remove some of the intelligent data processing functions from application modules and, instead, to provide generic intelligent service modules in the directory system being queried by the application modules, and (b) to better organize data items residing in the directory system so that they are more conveniently and more immediately accessible to a given intelligent service module function now residing in the directory system (as opposed to the inquiring application module). Preferably, an application module now need make only a single request to receive an intelligent directory service system response.

With this in mind, it will be noted that new independent method claim 139 is directed to a method for storage and retrieval of directory data in a directory system running on at least one processor having access to at least one data storage device and at least one communications network with interfaces to at least one application running on other processors having need of directory system services.

Claim 139 requires running a plurality of intelligent directory service modules as a part of the directory system. Those modules comprise at least one of three identified intelligent service functions: (a) identity management, (b) presence management, and/or (c) messaging management.

Claim 139 also requires that data objects used by the directory service modules be stored in respectively corresponding, different, organized logical segments of memory, each segment containing object attribute data needed by the corresponding directory service module to perform its intelligent service in response to an incoming request.

Claim 139 also requires that received directory service requests include an identification of the type of requested directory service (e.g., at least one of identity, presence and/or messaging services). The received directory service requests are then directed to the appropriate directory service module corresponding to the identified type of requested service.

Finally, claim 139 requires that responses be returned to incoming requests based on the output of at least one intelligent directory service module without requiring access of other object attribute data separately stored in association with another of the intelligent directory service modules.

Independent apparatus claim 182 will be seen to include similar requirements.

New dependent claims add many additional important features to the basic embodiment of the independent claims. For example, claim 140 requires that the intelligent directory service module include at least all three of the recited service modules. Claim 141 requires the intelligent directory service modules to provide customized virtual machines. Claim 142 requires that the intelligent directory service modules be embodied within solid state integrated circuits. Claim 143 requires the different organized logical segments of memory to be logical segments of memory providing a directory information tree (DIT). Claim 144 requires the DIT to be used to locate the logical segment of memory corresponding to the requested intelligent directory service and to access the object attribute data associated therewith.

The additional recitations of dependent claims 145-181 are self-evident from the above amendment, and the record need not be further burdened here to demonstrate that the new claim set includes numerous novel features nowhere suggested by Harvey.

The rejection of claims 1-8, 10-16, 32-37, 41, 78-79, 105-106, 124-126 and 132-137 under 35 U.S.C. §102 as allegedly anticipated by Harvey WO '147 is respectfully traversed.

In view of the above amendment cancelling all of these claims without prejudice or disclaimer, this ground of rejection has now been mooted and thus detailed comments supporting the traversal are not necessary at this time.

As detailed in Attachment 1 presented during the interview (and hereby incorporated by reference), the Harvey teaching can no longer provide an adequate directory system in view of modern requirements. The organization and operation of the Harvey system required more resources (in all respects, including hardware, software and response time) compared to the applicants' now claimed approach where intelligent service modules are found in the directory system itself – rather than requiring such intelligence, in effect, to be replicated many times in the many different application modules having need for such services. The applicants' novel architecture also drastically reduces the amount of data that needs to be communicated between the application modules and the directory service system.

In particular, the above-discussed requirements of independent claims 139 and 182 are nowhere to be found in Harvey.

The rejection of claims 18, 20-25, 27-30 and 39 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Murthy '039 is also respectfully traversed.

As before, since these claims have now been cancelled without prejudice or disclaimer, this ground of rejection has been mooted, and it is not necessary at this time to detail reasons for such traversal.

Furthermore, fundamental distinctions over Harvey have already been noted above for new independent claims 139 and 182, and Murthy does not supply those deficiencies.

The rejection of claims 9 and 38 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Sudia '616 is likewise respectfully traversed.

As before, this ground of rejection ahs been mooted by the above amendment. In addition, fundamental deficiencies of Harvey with respect to independent claims 139 and 182 are not supplied by Sudia.

The rejection of claim 19 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Cook '726 is similarly traversed. Once again, this ground of rejection has been mooted by the above amendment. Furthermore, the above-noted deficiencies of Harvey with respect to new independent claims 139 and 182 are not supplied by Cook.

The rejection of claims 77 and 130-131 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Langseth '980 is also respectfully traversed for similar reasons – including the fact that these grounds of rejection have been mooted by the above cancellation of these claims without prejudice or disclaimer. Furthermore, once again, fundamental deficiencies of Harvey with respect to independent claims 139 and 182 are not supplied by Langseth. Accordingly, it is not necessary at this time to discuss the additional features of the numerous dependent claims now presented.

Similarly, the rejection of claims 26 and 42 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Murthy/Cook is also traversed – and has been mooted by the above amendment.

The rejection of claims 43-45 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Hsu '664 is similarly traversed – and mooted by the above amendment.

The rejection of claims 46-48, 50-54, 56-57 and 59-61 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Kagan '681 is similarly traversed – and has also been mooted by the above amendment.

The rejection of claims 127-129 and 138 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Cotte '048 is likewise traversed – and has been mooted by the above amendment.

The rejection of claim 58 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Kagan/Langseth is also respectfully traversed for the same reasons.

The rejection of claims 62-66 and 70-73 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Kagan/Murthy is also respectfully traversed for the same reasons.

The rejection of claim 49 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Kagan in further view of Irwin '331 is similarly traversed – and has also been mooted by the above amendment.

The rejection of claim 55 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Byrne '382 is also traversed for the same reasons.

The rejection of claims 67 and 69 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Kagan/Murthy/Cotte is also traversed for the same reasons.

The rejection of claim 68 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Kagan/Cotte is also respectfully traversed for the same reasons.

The rejection of claim 102 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Roy '893 is similarly traversed for the same reasons.

The rejection of claims 103-104 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey in view of Dietterich '393 is also respectfully traversed for the same reasons.

The rejection of claim 107 under 35 U.S.C. §103 as allegedly being made "obvious" based on Harvey/Roy is also respectfully traversed for the same reasons.

Accordingly, this entire application is now believed to be in allowable condition, and a formal notice to that effect is earnestly solicited.

Respectfully submitted,

**NIXON & VANDERHYE P.C.**

By: Larry S. Nixon
Larry S. Nixon
Reg. No. 25,640

LSN:lef

901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100

ATTACHMENT 1

Wwite

# The Lloyd – Oliver
# Directory Service US Patent Application
# 10/705242

## Referred to as CADS:
## Composite, Adaptive Directory Services

The next generation directory technology

# To provide context for CADS patent application the important concepts are:

Name given to the directory service:
Composite, Adaptive Directory Services (CADS)

- "Composite" – because the Lloyd-Oliver directory service:
  - requires only one instance of each embedded function located in directory service (instead of prior art practice of embedding such functions in each of n directory service applications)
  - permits single directory access functionality rather than prior art requirement for multiple accesses to the directory service for a given function (located in each of n applications)
  - Stores object attributes for functions collectively in respectively corresponding different organized segments of memory thus permitting faster and one-time access to only those attributes needed for a given function
- "Adaptive" – because the Lloyd-Oliver directory service intelligently organizes specific sets of attributes for direct processing and optimization in concert with the embedded functions.

Typically embedded functions deal with user and device authorization, identity management, presence (events), security, messaging and service provisioning (multi objects).

Wwite

# Background

**The Lloyd-Oliver invention takes a view that existing LDAP directory service technologies need to evolve to be faster, distributed and more functional from an identity management, user service provisioning, presence and transactional messaging perspective, and it is this evolution that will change how all the other parts of a system are designed and deployed**

- Today large-scale directory systems are being developed with on-line user environment, security, end user service provisioning and identity management functions in mind. The difficulty with building such systems with a “basic directory” service as we have today (even a large scale one), is that the directory is coupled to many fragments of other system functions that each deal with (in their own way) user information, user validation, service oriented configuration files and so on.

- There are examples where these fragmented systems use up to 20 directory accesses just to validate a user and up to 80 accesses just to manage passwords. These are extreme cases, but we believe that these system designs are destined to fail if they are needed to support large scale ebusiness operations with many millions of customers.

- With the increase in personalization, complex authorization rules, location and device type issues to contend with the system would be much simpler if this authentication and authorization service was performed with <u>one</u> directory access using specialized processing  features of the directory.

- If a directory service is able to provide efficient and common user identity, presence management, service provisioning and user validation functions as well as an internal transactional messaging capability, ebusiness systems would be more efficient, faster, less costly and more scalable.

- Progressing this evolved directory functionality with an underlying capability that allows the critical parts of the directory information processing to be adapted and incorporated into integrated circuits (silicon), makes this evolution even more desirable.

## Older directory patents – e.g. the “Harvey” patent from Datacraft /Opendirectory designed the LDAP X.500 system using a RDBMS. At the time of its inception (by Lloyd) it was generally believed that RDBMSs could not be used to hold directory (object) information and named trees and have acceptable performance – hence the Harvey patent

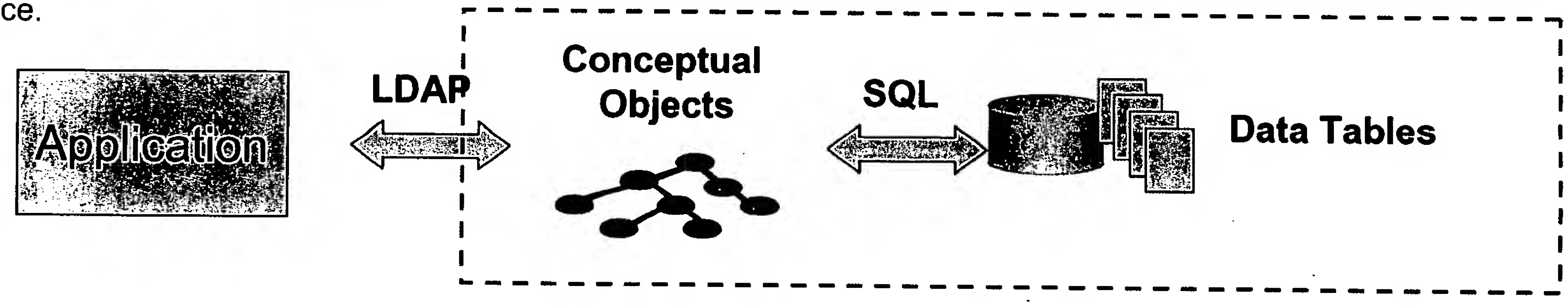
### Fundamental teachings of the Harvey patent are:

➢Standard LDAP/X.500 objects, attributes and name hierarchies are decomposed into SQL RDBMS tables where the data is held in a raw and normalized form and each table can be used individually or used with other tables to satisfy the standard LDAP/X.500 directory operations/services of Add, Modify, Remove, Rename, Search, Compare, etc.

➢The Harvey table data is cross referenced by EIDs and AIDs as to their directory name and attribute type and recomposed into objects in order to provide directory data responses.

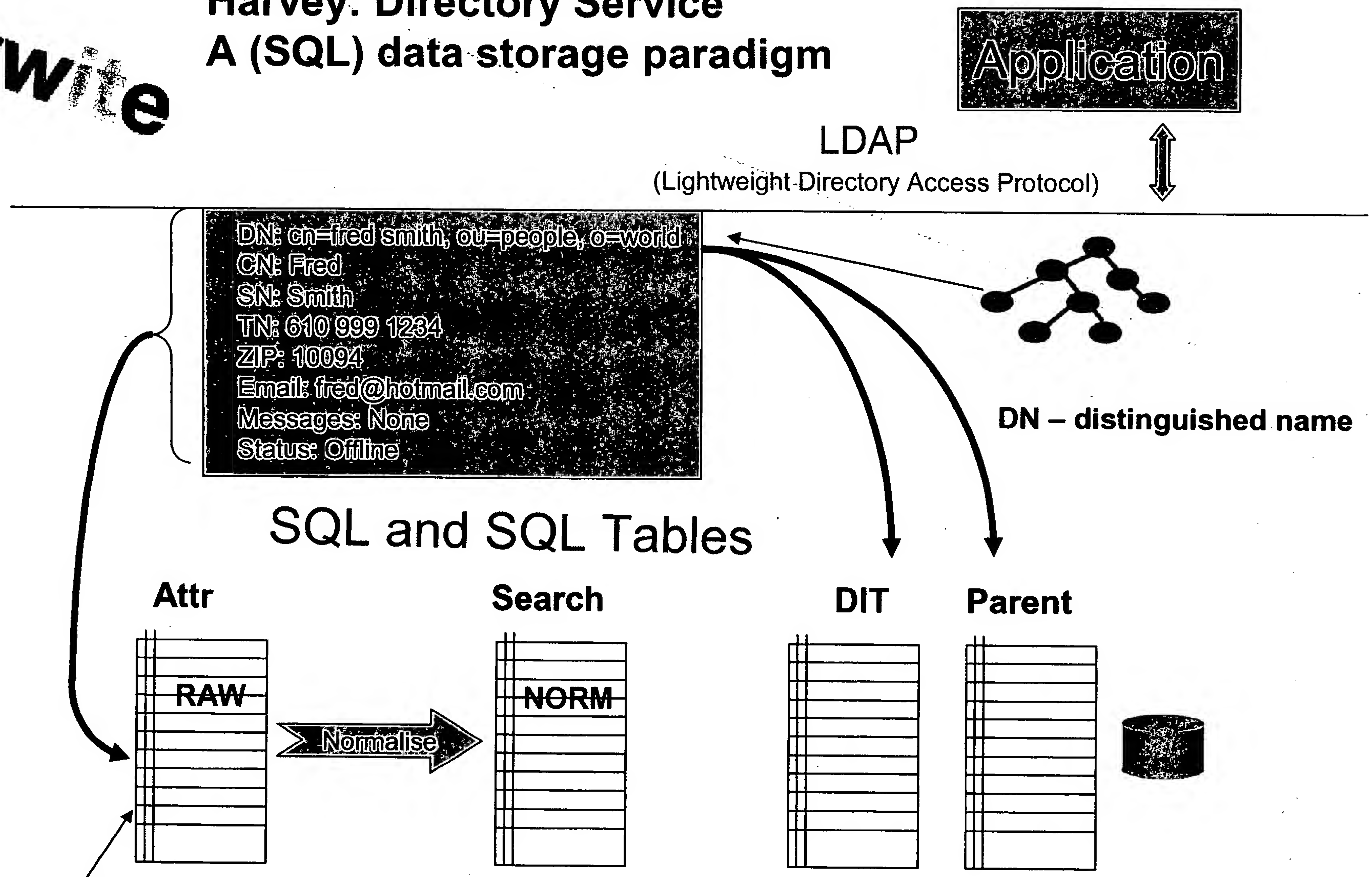
➢The Harvey patent describes processing LDAP service operations, decomposing the data and storing it in the tables using SQL, or retrieving from the tables via SQL data that is recomposed into directory objects and then sent back to the application.

➢The Harvey directory makes no determination as to the semantic applied to the names of the objects, the types of the attributes within the objects or how these are used by the application… it is strictly only a storage and retrieval device.



Wwite

# Harvey: Directory Service
# A (SQL) data storage paradigm



DN: cn=fred smith, ou=people, o=world
CN: Fred
SN: Smith
TN: 610 999 1234
ZIP: 10094
Email: fred@hotmail.com
Messages: None
Status: Offline

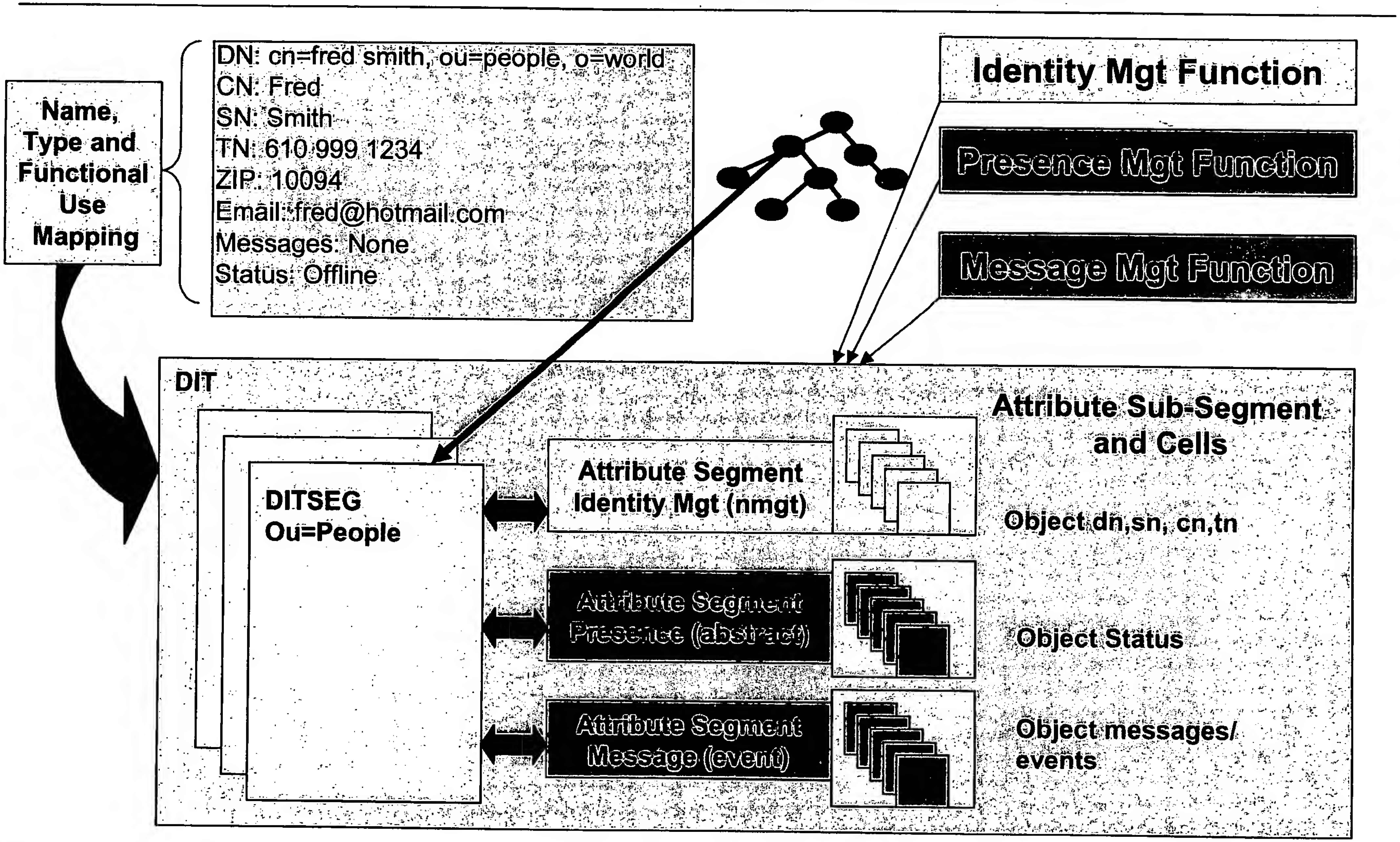
EID = Entry Id (Name/Level in dot form)
AID = Attribute Id (Type-> Syntax)

**Attribute/Search table size = (Objects X Object's Attributes). 1M objects , 10 attributes per object = 10M rows in Attribute table**
**DIT/Parent table size = 1M (number of objects)**

# Lloyd-Oliver: Directory Service (1)
## An information processing paradigm

Application

LDAP
(Lightweight Directory Access Protocol)



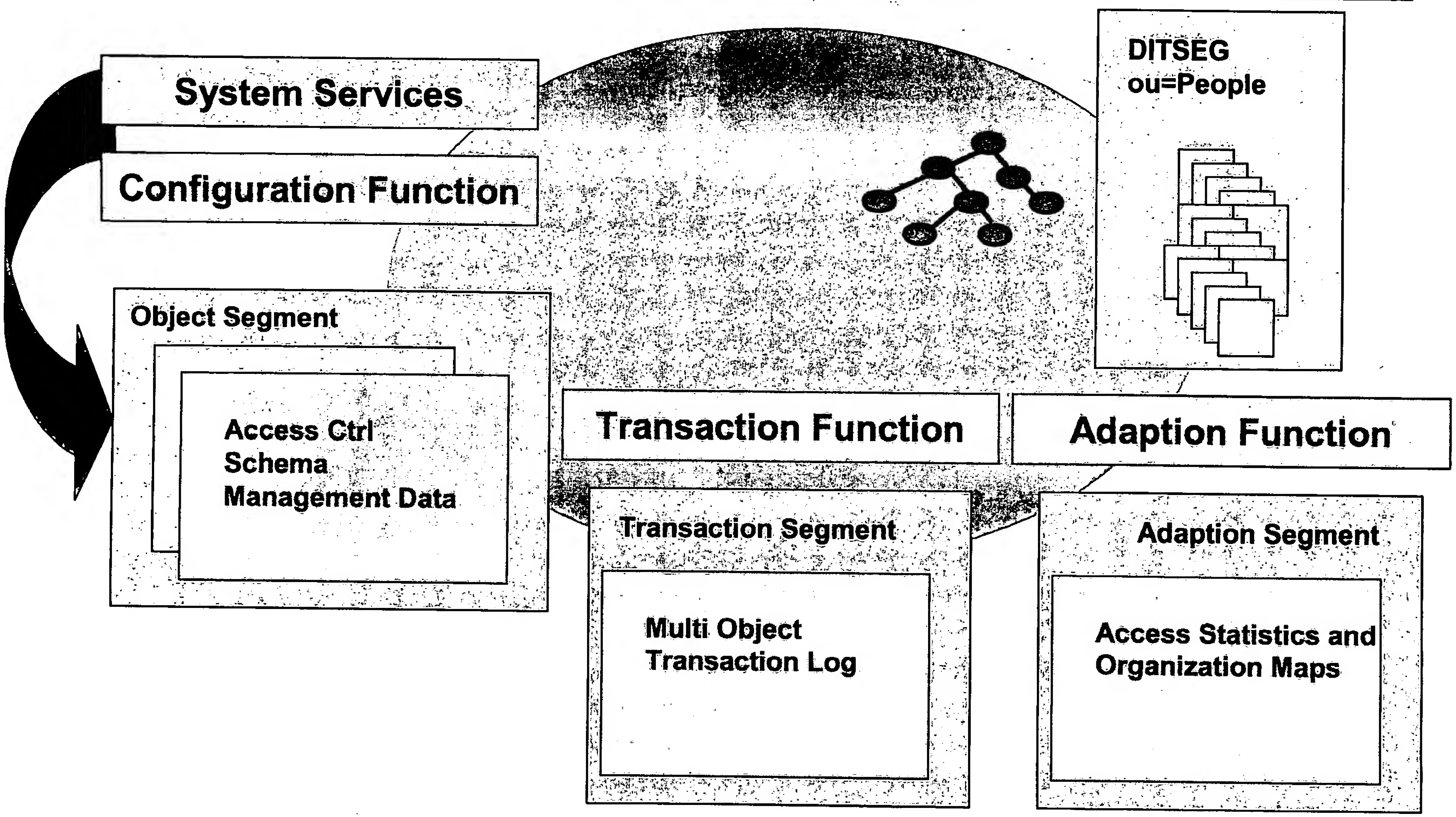
DN: cn=fred smith, ou=people, o=world
CN: Fred
SN: Smith
TN: 610 999 1234
ZIP: 10094
Email: fred@hotmail.com
Messages: None
Status: Offline

Name, Type and Functional Use Mapping
Identity Mgt Function
Presence Mgt Function
Message Mgt Function
DIT
DITSEG
Ou=People
Attribute Segment Identity Mgt (nmgt)
Attribute Segment Presence (abstract)
Attribute Segment Message (event)
Attribute Sub-Segment and Cells
Object dn,sn, cn,tn
Object Status
Object messages/ events

# Lloyd-Oliver: Directory Service (2)
## System and information management

Application

LDAP
(Lightweight Directory Access Protocol)



# The Lloyd-Oliver information processing and storage algorithm

- Allocate a DITSEG for a portion of the overall namespace..
  - If a directory has 1M users, 1M devices and 1000 services objects - and a DITSEG for each type, each storage segment is now a subset of the overall requirements.
- Under the DITSEG allocate an ATTRSEG for each intelligent function and its set of attribute properties such as identity management, presence, messages/events, images, security.
  - If a directory entry has 10 attributes and two are used for identity management/authorization, etc then each ATTRSEG only contains 2M attributes.
- This also permits one to allocate a file handle (a GUID) to each cell in each ATTRSEG that relates to the full DIT name (Distinguished Name) of the entry - and cache that cross reference.
  - When an entry is required by DN by an intelligent function– open the file: DITSEG name/ Function Name / GUID. C://DIT/ Ou=people/idm/aaa1234dirsvcs872.x

# The Lloyd design is novel and non-obvious

- The Harvey design is not extensible and has proven to have scaling and performance issues – it is <u>only</u> a directory data storage and retrieval algorithm.
- There is a difference in functionality of LDAP enabled applications that work with the Harvey design and those that work with the Lloyd-Oliver design – with advantages in the Lloyd-Oliver design.
- SQL/RDBMS is the sole basis for the Harvey design - which is <u>not</u> used in the Lloyd design
- That there is a storage form distinction as well:
  - Harvey: RDBMS Tables/Rows (Tables are singular)
  - Lloyd-Oliver: Segments, Subsegments and Cells (all can be plural)
- Table/Segment names and functionality: Harvey is totally different to Lloyd-Oliver
- Data forms and indexes:
  - Harvey is EID and AID:
  - Lloyd-Oliver uses neither.  Functional attributes in ATTRSEGS are contained in DITSEGS
- Attribute organization and management:
  - Harvey uses AIDs and stores data in table (raw and normalized form).
  - Lloyd organizes and processes attribute types according to embedded system functions and stores types and values in different segments of organized memory.
- Embedded intelligent system functions: Are not present in Harvey design – at all. But advantageously present in Lloyd-Oliver